

Csoport neve: The Racoons

Feladat sorszáma: H_3

Feladat címe: Szoftver specifikáció

Clean Paws Library elektronikus könyvtár

Gyakorlatvezető:

Árvai László

Csoport tagok:

Árvai László	TANKÖR	laszlo.arvai@uni-miskolc.hu
Kovács Dániel	F9Y7TW	kovacsd435@gmail.com
Mészáros István	WZZTLL	mesziist@gmail.com
Mester Dániel	FZIE3D	danielmester22@gmail.com
Fónagy Zeus Vilmos	G6HV2W	fonagyzeusz02@gmail.com
Kiss Tamás László	WN6YVX	tamaskiss2003@gmail.com

BEADÁS DÁTUMA

Történet

Dátum	Verzió	Leírás	Szerző
2023.10.28	0.1	A feladat elkezdése	Mindenki
2023.10.29	0.12	Első pontok feltöltése	Mindenki
2023.10.30	0.13	A rendszerfunkciók feltöltése	Mindenki
2023.11.01	0.15	Maradék, üresen maradt pontok feltöltése	Mindenki
2023.11.05	0.19	A pontok csiszolása	Mindenki
2023.11.09	1.0	A feladat beadása	Kovács Dániel

1. Bevezetés

Az internet terjedésének köszönhetően bárki bármilyen információt szerezhet a világ bármely pontjáról. Régebbi időkben, az információk többségét könyvtárakból tudhatták meg az emberek ezért cégünk egy olyan szoftvert kínál fel a könyvtáraknak, amely segítségével búcsút mondhatnak a papíralapú vagy digitális táblázatokban tárolt leltáraiknak.

A Clean Paws Library segítik a könyvtárosokat és könyvtárakat a könyvek és egyéb könyvtári anyagok hatékony kezelésében és nyilvántartásában. A programunk célja az, hogy segítse a könyvtárakat a könyvek katalogizálásában, a kölcsönzések és visszavétel nyomon követésében, a könyvtárhasználók nyilvántartásában, és egyéb adminisztratív feladatokban.

A program lehetővé teszi a könyvek és egyéb könyvtári anyagok részletes adatainak rögzítését, mint például cím, szerző, kiadó, ISBN, besorolás stb.

Az olvasók könnyen kereshetnek a könyvtár katalógusában, és kiválaszthatják a kölcsönzésre szánt könyveket és a program nyomon követi a kölcsönzéseket, a határidőket és a visszavételeket, meggyorsítva az ügyintézés. A végfelhasználók megújíthatják olvasójegyüket, valamint rendezhetik a késedelmi díjakat online is.

Az olvasni vágyókat egy egyszerű könnyen kezelhető webes felület fogadja, így a velünk szerződött könyvtáraknál lehetőség nyílik könyvek kölcsönzésére online felületen, valamint házhozszállítással segítünk olvasóinknak kedvenc könyvük elolvasásához.

A könyvtárak részére biztosítjuk a tároláshoz szükséges szervereket ezért egyéb költség nem terheli őket. A szervereinken diszkréten kezeljük a könyvtár, illetve az olvasók adatait így gondoskodva a könyvtári adatok és felhasználói adatok védelméről.

2. Áttekintés

Ez az alkalmazás azért készült, hogy a manapság hanyatló könyvtár infrastruktúrák és az olvasás köztudatba való visszahozásában segít. Manapság az emberek nem nagyon mennek könyvtárakba, ugyanis, ha az nincs az alagsorban egy, hosszabb utat kéne megtenniük, mint hogy fellépjenek az internetre és megszerezzék onnan elektronikusan. de ez az a szoftver segít nekik, hogy a könyvtárak jussanak el a házaikba, még hozzá papír alapú könyvekként, ahogy a jóisten is teremtetten őket.

A szoftverünk nem csak a kedves olvasók terhein könnyít, hanem a könyvtárakén is, ugyanis a papír alapú dokumentációt digitálisra cseréljük. Ezáltal sokkal könnyebb lesz az adminisztrációs munka. Illetve a könyvtári gépeket is kíméli, mivel az alkalmazás a felhőben fut nagyrészt. (Persze nagyon szigorú adatvédelemmel majd)

A rendszer minden egyes felhasználójáról vezet profilt, akik különböző mélységekben használhatják a programunk funkcióit. Például az egyszerű felhasználó pénzt tölthet föl, könyvek kölcsönzési idejét hosszabbíthatja, olvasójegyek beszerzése, akár olcsóbban is, hírlevelekre

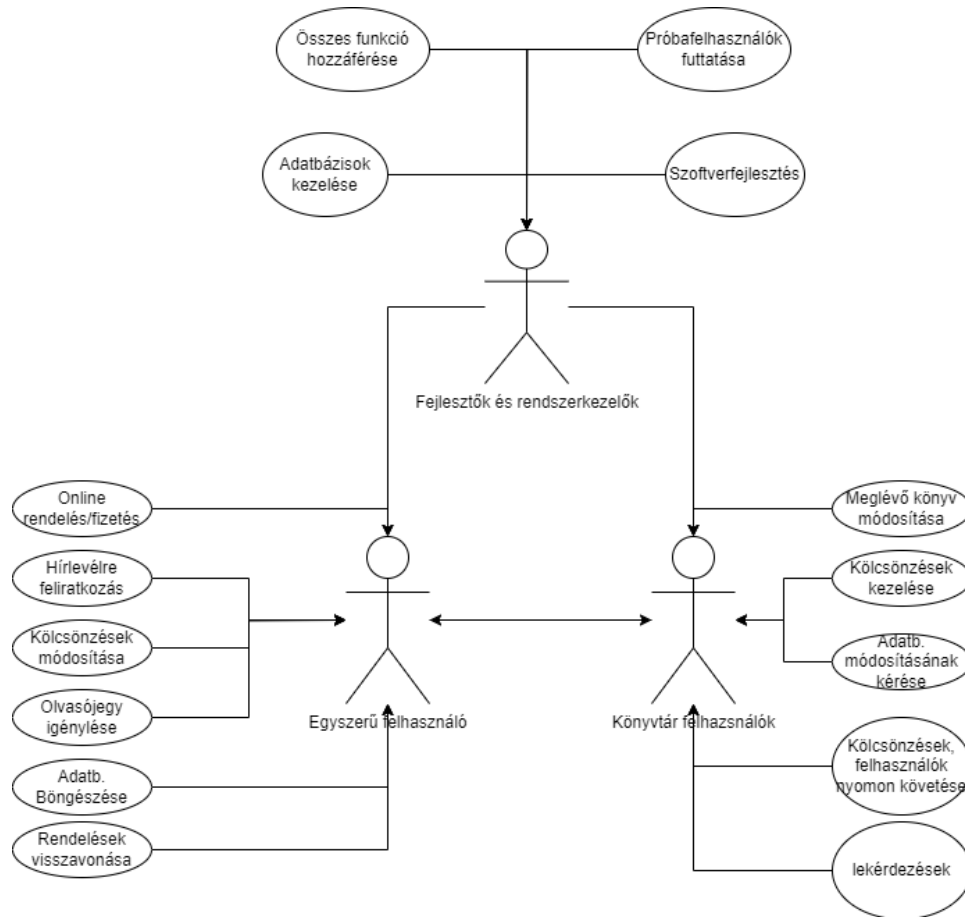
feliratkozás a legfrissebben beszerezett könyvekről érkező értesítések nevében, továbbá szerkeszthetik a profiljukat, hogy nekik, itt nem profilképre gondolok, mert minek is az, hanem mondjuk e-mail, lakcím vagy telefonszám módosításokra. A könyvtárosoknak kicsit nagyobb hozzáférésük van: ők már látják a teljes adatbázist, de szerkesztéséhez az erre kijelölt szerveknek kell kérvényt írni. A saját profilt ők is tudnak szerkeszteni, de itt nem olyan értelemben, mind a sima felhasználóknak. Ők a könyvtár profiljához férnek hozzá a weboldalon/appban. Akciókat írhatnak ki, látják a kölcsönzések listáját (az összes felhasználójét), módosíthatják a könyvtárukban elérhető könyvek listáját a hatalmas adatbázis alapján.

Valamint vezet egy hatalmas adatbázist (emiat egy 2 terrabájtnyi tárhellyel rendelkező szervert kell majd beszerezni, hogy minden simán működjön, de ezt persze csak akkor, ha a programunk nagyon befut, kezdetleg elég akár egy terrabájtos is), ami a könyvtárakban (akik szerződtek velünk) fellelhető összes könyvet tartalmazza, és a más könyvtáraknak csak össze kell vetni a saját adatbázisukkal, hogy fellelhető náluk ez a könyv, és csupán egy igen/nem bitet kell módosítaniuk az ügyben, hogy ezt jelezzék is a külvilágnak. De minek is kell ez az adatbázis? A keresések megkönnyítése érdekében, ugyanis a felhasználó rákeres egy könyvre az erre kitalált kereső felületen, akkor nem a könyvtárak adatbázisán fog egyesével végig rohanni a program, hanem az egyetemesen, és kiírja azok könyvkölcsönző üzemek nevét, ahol ennek a könyvnek az elérhetősége igazra van állítva.

Az alkalmazásnak egy könnyen kezelhető, gyors felülettel kell rendelkeznie (szövegek helyett inkább képek, kevés gomb, maguktól értetődő nevek, gomb-feliratok), amit a tizenévesektől az aggastyánoknak is tudni kell kezelni, ugyanis az öregeknek van a legtöbb idejük arra, hogy könyveket olvassanak. Ezért még egy sűgőt is implementálni kell, mert a vannak olyanok, akik egy egyszerű felületet sem tudnak segítség nélkül kezelni.

A szoftver platform független, tehát operációs rendszertől függetlenül alkalmazható. Ezt a tulajdonságot nem sok mai alkalmazás mondhatja el magáról, mind amellet, hogy napjainkban egyre jobban terjednek a platform független alkalmazások. Ezáltal biztosítva nagyfokú kompatibilitást a ma már meg lévő rendszerekkel. Ezt a CLP szoftverünk esetében úgy biztosítjuk, hogy JAVA nyelven implementáljuk. Emiat egyetlen megkötés, hogy rendelkezésre kell állnia egy webservernek, amely lehetővé teszi a szoftver futtatását. A klienseknek csupán Internet eléréssel és egy webböngészővel kell rendelkezni.

3. A rendszer funkciói



A diagrammon található akceptorok

Felhasználók:	Itt az egyszeri felhasználókra gondolunk, mint Mariska néni a fotelból. Nekik lesz a legkevesebb belelátása a rendszer mögött történő adatbázis-műveletekre, ők csak a funkciókat tudják beindítani, már amihez van hozzáférésük.
Könyvtári felhasználók	Ők a könyvtáranként kijelölt ember/emberek, akik a saját létesítményük profilját tudják szerkeszteni a weblapon/appon (ha lesz). Illetve látják az összes felhasználót, akik valamiféle kapcsolatba kerültek könyvtárunkkal, de ezeket is csoportosítva. (például olvasójegy vásárlása.) A saját adatbázisukat monitorozva figyelhetik a központi adatbázist, és módosíthatják a saját

	adatbázisuk alapján a könyvek elérhetőségét (ez automatizálva lesz, a könyvtárosoknak csupán egy “egybevetés” gombra kell kattintaniuk. De persze ez majd manuálisan is végezhető azok kedvéért, akik inkább ezt az utat választják)
Fejlesztők	Itt nem csak azokra gondolunk, akik a szoftver újabb és újabb verziót készítik, hanem itt mindenki, aki a háttérből irányítja a gépezetet, csak így egyszerűbb róluk beszélni, hogy egy kalap alá vesszünk mindenkit. Az ő feladatuk a bugok elhárítása, ha fellépnek, új verziók készítése, kritikus rendszerhibák deadlockok, megoldása és a rendszer hibák utáni visszaállítása. Az adatbázis (a nagyobb) közvetlen kezelése, ennek módosítása, ha könyvtári adatbázisok módosulnak (ez is inkább automatizálva lesz.)

USE-CASE-ek

Egyszerű felhasználók:

- Online kölcsönzés/fizetés
- Hírlevélre való feliratkozás
- Kölcsönzési idő hosszabbítása
- Hozzáférés az adatbázishoz (csupán böngészés)
- Rendelések visszavonása

Online kölcsönzés/fizetés:

A felhasználó ennek a funkciónak a segítségével tud a házához könyveket kölcsönözni, illetve online fizetni értük, az olvasójegyekért is így tud fizetni. Ezt a funkciót, egy adott könyvtár adott könyvének a fülére, és az ott található fizetés gombra kattintva lehet elérni. Az olvasójegyet ugyanígy, csak ott az olvasójegy igénylése gombra kattintva.

Prekondíciók (előfeltételek): A felhasználó regisztrált, illetve nincs semmiféle szabályzat megszegése miatt felfüggesztve. Megadott bankkártya adatot, illetve lakcímet, ha házhoz kívánja rendelni. A könyvtárnak rendelkeznie kell példánynak a rendelni kívánt könyvből. Illetve nincs ugyanennél a könyvtárnál ugyanilyen típusú olvasójegye (ha azt kívánna venni).

Postkondíciók (következmények): A felhasználó és a könyvtár között létrejön a kapcsolat, amely a rendelés adatait tartalmazza. A felhasználó bankkártyájáról eltűnik a fizetendő összeg. Amennyiben olvasójegyet vett: megjelenik a profilján az olvasójegyek fül alatt, és a könyvtár is kap értesítést, illetve módosul azon felhasználó adatbázis-helyének az olvasójegy tulajdonsága. Amelyről a könyvtár is kap értesítést.

Main flow (normális működés): Megjelenik egy kis ablak, ahol meg tudjuk a lakcímet, illetve a fizetési adatokat, majd a rendelés gombra rákattintva a rendelés megjelenik értesítésként a könyvtár profilján. Olvasójegynél ugyanez, csak ott még név, illetve egyéb adatokat is kérnek, ami a könyvtár olvasójegy-kiadásához szükséges.

Alternate Flows (Alternatív esetek): Akkor következnek be, ha a felhasználó úgy kíván házhoz rendelni, hogy nincs megadva lakcíme. Úgy akar fizetni, hogy nincs elég pénze a tranzakció végrehajtásához. Félreütött lakcím vagy bankkártya adatok esetén is bekövetkezhethet.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Hírlevélre való feliratkozás:

A felhasználó ennek a funkciónak a segítségével kaphat értesítéseket egy általa kiválasztott könyvtár profile-módosulásairól. Ezt az adott könyvtár publikusan látható profilján teheti meg a "Feliratkozás a hírlevélre" gombra kattintva.

Prekondíciók (előfeltételek): A felhasználó regisztrált, illetve nincs semmiféle szabályzat megszegése miatt felfüggesztve. Illetve a könyvtár se szűnt még meg.

Postkondíciók (következmények): A felhasználó értesítéseket fog kapni azokról a módosításokról a könyvtár profilján, amikről ő is tudósítani akar (pl: akciók kiírása, új könyv érkezése)

Main flow (normális működés): Gombra rákattintva a felhasználó adatbázisában elfoglalt helyén a feliratkozások tulajdonság bővül a könyvtár ID-jével, ami arra szolgál, hogy a hírlevél küldő funkciója a könyvtárnak megtalálja azokat az embereket, akik fel vannak iratkozva.

Alternate Flows (Alternatív esetek): Amennyiben se a könyvtár se a felhasználó nem rendelkezik profillal (habár ilyen soha nem fog bekövetkezni.)

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Kölcsönzési idő hosszabbítása

A felhasználó ennek a funkciónak a segítségével tudja majd hosszabbítani a kölcsönzési idejét egy adott könyvnek. Ezt a kölcsönzések fülön teheti majd meg, az adott könyv kölcsönzési hosszabbítása gombra kattintva. A fizetési értéket az adott könyvtár szabja meg.

Prekondíciók (előfeltételek): A felhasználó regisztrált, illetve nincs semmiféle szabályzat megszegése miatt felfüggesztve. Az adott könyvtárnál legyen egy függő kölcsönzése.

Postkondíciók (következmények): A kölcsönzési idő módosul a kívánt értékre.

Main flow (normális működés): A gombra rákattintva felugrik egy ablak, amellyel a kölcsönzési idő új értékét lehet megadni, majd a jóváhagy gombra kattintva véghez is vinni azt (amennyiben a felhasználó nem rendelkezik az adott könyvtárnál olvasójeggyel, akkor felugrik egy fizetés ablak is).

Alternate Flows (Alternatív esetek): Akkor következnek be, ha a felhasználó úgy kíván házhoz rendelni, hogy nincs megadva bankkártya adat. Úgy akar fizetni, hogy nincs elég pénze a tranzakció végrehajtásához.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Hozzáférés az adatbázishoz (csupán böngészés):

A felhasználó ennek a funkciónak a segítségével tud könyveket keresgélni a teljes adatbázisban. Ezt a funkciót a "keresés" -bar-on tudja majd elérni.

Prekondíciók (előfeltételek): A felhasználó regisztrált, illetve nincs semmiféle szabályzat megszegése miatt felfüggesztve.

Postkondíciók (következmények): A felhasználó egy listát kap a fő adatbázisból, amelyek valamilyen úton módon illeszkednek a név tulajdonságra a könyveknél.

Main flow (normális működés): A kereső felületbe történő begépelés után egy lekérdezés történik az adatbázisban, amely a begépelte karakterek szerint gyűjti ki a könyveket, és továbbítja a felhasználónak.

Alternate Flows (Alternatív esetek): Akkor következnek be, ha a felhasználó olyan könyvet keres, amely nincs benne az adatbázisban.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Rendelések visszavonása:

A felhasználó ennek a funkciónak a segítségével tudja a függőben lévő rendelését visszavonni. Ezt a fizetés után felugró ablakban teheti meg a “rendelés visszavonása gombra kattintva”.

Prekondíciók (előfeltételek): A felhasználó regisztrált, illetve nincs semmiféle szabályzat megszegése miatt felfüggesztve. Van rendelése, amelyet vissza kíván vonni.

Postkondíciók (következmények): A felhasználó és a könyvtár között megszűnik a kapcsolat, amely a rendelés adatait tartalmazza. A rendelés törlődik az adott könyvtár listájából.

Main flow (normális működés): Megjelenik egy kis ablak miszerint a törlés sikeres volt, és a rendelés megszűnik létezni.

Alternate Flows (Alternatív esetek): Akkor következnek be, ha felhasználó nem adott le rendelést, illetve már megkapta a könyvet/olvasójegyet.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Könyvtári felhasználók:

- Meglévő könyv módosítása
- Kölcsönzések kezelése
- Adatbázis módosításának kérése
- Kölcsönzések/felhasználók nyomon követése
- Egyéb adatok lekérdezése

Meglévő könyv módosítása:

Amennyiben új könyv kerül fel az adatbázisba (könyvtári oldalon) a fő adatbázis is frissül, hogy az a könyvtár szintén rendelkezik ebből a példányból. Ez a könyvtári adatbázis frissítése után következik be.

Prekondíciók (előfeltételek): A könyvtár regisztrált, illetve nincs felfüggesztve a szabályzatok megsértése miatt. Ténylegesen (fizikálisan) kell rendelkeznie az adott könyvből.

Postkondíciók (következmények): A fő adatbázisban módosul az adott könyvhöz a “megtalálható” tulajdonság. Megjelenik a könyvre való kereséskor a listában a létesítmény neve.

Main flow (normális működés): Frissítés után egy lekérdezés, illetve egy egyeztetés indul el a fő adatbázisban. Ahol megnézi a szoftver, hogy a módosított mezők mely könyveket érintik és átírja azok értékeit.

Alternate Flows (Alternatív esetek): Akkor következnek be, ha a könyvtár olyan könyvet kíván felvinni, amellyel nem is rendelkezik. Ez súlyos szabálysértés, és felfüggesztéssel jár. Illetve véletlenül kétszer kerül be az adat a könyvtár adatbázisába. A felvitt könyv nem szerepel a fő adatbázisban, ekkor jön majd a fő adatbázis módosítása.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Kölcsönzések kezelése:

A könyvtáraknak lehetőségük van látni a náluk éppen folyamat alatt történő kölcsönzéseket. Illetve módosíthatják azokat. Ezt az aktív ügyfelek fülön tudja elérni.

Prekondíciók (előfeltételek): Az adott könyvtár profilja regisztrálva van, illetve nincs felfüggesztve a szabályzat megsértése miatt. Az adott felhasználó, akinek módosítani akarják az idejét, ténylegesen kölcsönzött abból a létesítményből. Valamint szükséges egy valid indoklás, hogy miért is történik a változtatás.

Postkondíciók (következmények): A felhasználó adott kölcsönzési ideje: nő, csökken törlődik.

Main flow (normális működés): Megjelenik egy kis ablak, ahol a könyvtár a módosításokat kérvényezheti, illetve egy szövegdoz, ahova az indokot gépelheti, majd ezt elküldi az adott ügyintézőnek.

Alternate Flows (Alternatív esetek): Akkor következik be, ha az indoklás nem validálható. Illetve valamilyen túl nagy, illetve negatív számra módosul az érték.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Adatbázis módosításának a kérése:

Miután a könyvtár adatbázisa frissült olyan könyvekkel, amelyek eddig nem voltak elérhetőek a fő adatbázisban, vagy esetleg a módosítás miatt eltűnt a z utolsó fellehető példány a regisztrált könyvtárakból, akkor egy automatikus módosítási kérelem irányul az össz-adat módosítására.

Prekondíciók (előfeltételek): Az adott könyvtár profilja regisztrálva van, illetve nincs felfüggesztve a szabályzat megsértése miatt. A változtatás validálható (a könyvtár tényleg rendelkezik abból a könyvből.)

Postkondíciók (következmények): A könyv elérhető válik az összesített adatbázisban.

Main flow (normális működés): Módosítás után, ha még nincs a teljes adatbázisba a könyv, vagy kikerül onnan mert már senkinél sem található, akkor módosul az adott műveletnek megfelelően.

Alternate Flows (Alternatív esetek): Ha valamilyen oknál fogva újra fel akarná vinni a könyvet az adatbázisba (mert mondjuk nem találta). Hibás című könyvek felvitele.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Kölcsönzések/felhasználók nyomon követése:

A könyvtáraknak lehetőségük van látni a náluk éppen folyamat alatt történő kölcsönzéseket. Illetve azokat a felhasználókat, akik náluk váltottak olvasójegyet. Szintúgy az aktív ügyfelek felületen elérhető.

Prekondíciók (előfeltételek): Az adott könyvtár profilja regisztrálva van, illetve nincs felfüggesztve a szabályzat megsértése miatt. A felhasználó szintúgy létezen, de lehet felfüggesztés alatt.

Postkondíciók (következmények): Egy automatikusan frissülő lista, amelyet könnyű átlátni ezáltal a könyvtárosok gyorsabban tudnak műveleteket végrehajtani a felhasználókon.

Main flow (normális működés): Amint valamilyen felhasználó művelet zajlik, amely érint egy könyvtárat akkor az adott profilban módosítja a megfelelő információkat.

Alternate Flows (Alternatív esetek): Akkor következik be, ha a valamilyen oknál fogva nem frissül a felhasználó-lista.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Egyéb adatok lekérdezése:

Ennek segítségével látják könyvtár-profil kezelő dolgozók, hogy mennyire sikeres, illetve sikertelen volt a munkahelyük a CPL-en a regisztrálásuk óta. Ezt majd a statisztikák alatt tekintheti meg az ezzel foglalkozó személy. A lista automatikusan frissül, de kérésre lehet majd frissíteni egy külön gombbal a fülön.

Prekondíciók (előfeltételek): Az adott könyvtár profilja regisztrálva van, illetve nincs felfüggesztve a szabályzat megsértése miatt.

Postkondíciók (következmények): Egy jól követhető adatbázis-szerűség, ahol a bevételt, kölcsönzött könyvek mennyiségét hónapra lebontva, felhasználók, akik itt rendelkeznek olvasójeggyel és abból milyen fajtaival, illetve a hírlevelükre feliratkozottak száma.

Main flow (normális működés): Amikor a Postkondíciókban említett adat frissül valahol az adatbázisokban, vagy kifizetés történt a rendelésekkel, akkor ez a funkció módosítani fogja az adott könyvtár profilján található azon értéket. Ez a funkció egy órát fog használni, ami körülbelül óránként fogja lefuttatni a programkódot ezáltal is kímélve a rendszert.

Alternate Flows (Alternatív esetek): Ha az óra elakad valahol, vagy túl sok változás következik egy óra alatt és belerokkan a rendszer

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver

meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Fejlesztők:

- Adatbázisok kezelése
- Próbafelhasználók futtatása
- Hozzáférés az összes eddigi funkcióhoz

Adatbázisok kezelése:

A fejlesztőknek van külön fejlesztői accountjuk, ahonnan minden olyan funkciót elérnek, amelyeket más felhasználók nem tudnak, ilyen például a fő adatbázis szabadon szerkesztése, egész láthatósága (akármilyen lekérdezés írása, adatok kedv szerint törlése, felvitele, persze ha van rá megfelelő indok).

Prekondíciók (előfeltételek): A műveletet végrehajtó egyénnek kötelező, hogy fejlesztői fiókja legyen.

Postkondíciók (következmények): A könyvtárak, felhasználók, és az összesített könyv adatbázis szabadon való szerkesztése.

Main flow (normális működés): A fejlesztő olyan műveletek végezhet az adatbázissal amikor csak akarja, persze megfelelő adatbázis szerkesztők, illetve engedélyeknek a birtokában.

Alternate Flows (Alternatív esetek): Valamilyen oknál fogva nincs jogosultsága a fejlesztőnek az adatbázis szerkesztésére.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

Próbafelhasználók futtatása:

A fejlesztőknek lehet arra lehetőségük, hogy magán az accountjukon belül különböző felhasználó-látásmódra váltsanak.

Prekondíciók (előfeltételek): A műveletet végrehajtó egyénnek kötelező, hogy fejlesztői fiókja legyen.

Postkondíciók (következmények): A dolgozó személy különböző szemszögekből látja a programot, így könnyebben meg majd a bugok javítása, illetve a fejlesztés.

Main flow (normális működés): A mód aktiválása alatt a személy látja, hogy mit látna egy egyszerű felhasználó, vagy egy könyvtári dolgozó a szoftverünkben. Mindezt egy hamis felhasználói fiókkal érjük el.

Alternate Flows (Alternatív esetek): Valamilyen oknál fogva a fejlesztő nem tud, illetve átváltani a felhasználói nézetekből.

Exception Flows (Kivételes esetek): Olyan hibák bekövetkezése, mely a rendszer használata során váratlanul, a rendszer, illetve a felhasználó hibáján kívül következik be, melyeket a szoftver tervezőinek kell megoldania. Ezen események felmerülhetnek a szerver meghibásodásából eredően, illetve egy esetleges áramkimaradás esetén, hacker támadás esetén, valamint a szerver és a kiszolgáló gép közötti kommunikációban keletkező problémákból.

3.1 A hardverrel szemben támasztott követelmények

- Grafikus megjelenítés támogatása (hogyan lássuk a weboldalt)
- Egy x86-os processzor, hogy a weboldal simán fusson.
- A memória mennyisége elég legyen a futtatási környezet megteremtéséhez (ez számítógép-típusonként nagymértékben változhat)
- A merevlemez 100MB szabad terület
- Hálózati kártya a weboldal eléréséhez

3.2 Szoftverszintű követelmények

- A weboldal futtatása engedélyezve legyen a vírusirtó/böngészőn belül.
- Megfelelő szerverre való csatlakoztatás
- Java programok futtatásának lehetősége
- Elérhető MySQL adatbázis
- HTML és CSS futtatási lehetőség

3.3 Felhasználó számára támasztott követelmények

- Egy időben több felhasználó is használhatja az adatbázist, a korlátozások figyelembevétele mellett
- Egy időben több felhasználó is használhatja rendszert
- Az elvesztett, elfelejtett jelszavak nem pótolhatók, annak gondos kezelése a felhasználó felelőssége (még).
- Szeresse a könyveket!

4. Használhatóság

A rendszer használata könnyen, gyorsan elsajátítható minden felhasználói korcsoportnak az egyszerű felhasználóbarát grafikus felhasználói felületnek (GUI) köszönhetően. Jól átlátható, hasonló elrendezést használunk, mint a már létező, eddig használhatott könyvtári adatbázis rendszer alkalmazások a könnyű átállást elősegítve.

Törekszünk arra, hogy a lehető leggyorsabb válaszidőt nyújthassuk felhasználóink számára folyamatos karbantartással és fejlesztésekkel.

Alkalmazásunkban található egy súgó, illetve egy gyakori kérdések (FAQ) menüpont, amelyek segíthetik a felhasználókat a szoftver használatát illetően. A visszajelzések alapján könnyebben felmérhetők a felhasználók igényei, elősegítik a további fejlesztéseket. A betanulás segítésére és az esetleges elakadások megelőzésének céljából a szoftverhez online elérhető egy részletes felhasználói dokumentáció, melyben minden funkció, beállítás megtalálható.

5. Megbízhatóság

Egy könyvtár adatbázisrendszer szoftverrel szemben támasztott megbízhatósági követelmények a következők lehetnek:

1. **Adatintegritás:** Az adatbázisrendszernek biztosítania kell az adatok integritását, vagyis az adatoknak mindenkor helyesnek és érintetlennek kell lenniük. Ne lehessen véletlenül vagy szándékosan módosítani, vagy törölni az adatokat.
2. **Magas rendelkezésre állás:** A könyvtár adatbázisrendszernek folyamatosan elérhetőnek kell lennie a felhasználók számára. Bármilyen hibajavítást vagy karbantartást úgy kell tervezni, hogy minimális legyen a leállás.
3. **Teljesítőképesség:** A rendszernek képesnek kell lennie megbirkózni a növekvő adatmennyiséggel és a megnövekedett felhasználói igényekkel anélkül, hogy a teljesítmény érezhetően csökkenne.
4. **Biztonság:** Az adatbázisrendszert megfelelően kell védeni a jogosulatlan hozzáférés, adatlopás vagy adatvesztés ellen. A felhasználók hozzáférése csak jogosultaknak és kijelölt személyeknek kell engedélyezni.
5. **Automatikus mentés és helyreállítás:** Az adatbázisrendszernek rendszeres időközönként automatikusan kell mentést készítenie az adatokról, és képesnek kell lennie az adatok gyors helyreállítására hibás működés vagy adatvesztés esetén.

6. **Kompatibilitás és szabványosság:** A rendszernek kompatibilisnek kell lennie más szoftverekkel és eszközökkel, valamint betartania kell az adatbázis szabványokat és irányelveket.
7. **Riasztás és hibajelzés:** A rendszernek képesnek kell lennie riasztások és hibajelzések küldésére, hogy a rendszergazdák azonnal értesüljenek az esetleges hibákról vagy rendellenességekről.
8. **Skálázhatóság:** A rendszernek lehetővé kell tennie a könnyű skálázhatóságot, hogy az adatbázist az igények növekedése esetén bővíteni lehessen.
9. **Felhasználói támogatás:** Az adatbázisrendszernek fel kell kínálnia dokumentációt és támogatást a felhasználóknak a használatához és a problémák megoldásához.

Ezek a megbízhatósági követelmények segítenek abban, hogy a könyvtár adatbázisrendszer a lehető legmegbízhatóbb és hatékonyabb legyen, és az adatok védelme és hozzáférhetősége a lehető legjobban biztosított legyen.

A rendelkezésre állás mértékét a következő mennyiségekkel jellemezhetjük:

- **MTFF (Mean Time to First Failure):** Az üzembehelyezést követő első meghibásodás várható ideje. Sok gyakorlati esetben az első hibaig várható idő különbözik a későbbi javításokat követő hibamentes működési időtől, ezért szokás az MTFF-et megkülönböztetni a következő jellemzőtől.
- **MTTF (Mean Time To Failure):** Javítás utáni üzembe helyezéstől a következő meghibásodásig eltelt várható idő. Tapasztalati alapon általában a javítások utáni hibamentes működés ideje nem, vagy csak kis mértékben függ attól, hogy hányszor javították már a rendszert.
- **MTBF (Mean Time Between Failures = MTTF + MTTR):** Egy működési és állási fázis várható ideje, azaz két meghibásodás között várhatóan eltelt idő. Gyakorlatilag a rendszer ciklusideje a meghibásodások szempontjából.
- **MTTR (Mean Time To Repair):** A hiba észlelésének + a hiba meghatározásának + a javításnak a várható ideje.
- **Rendelkezésre állási tényező (Availability):** A rendelkezésre állási tényező a helyes működés ideje a teljes üzemidőhöz viszonyítva. Azt fejezi ki, hogy egy véletlenszerű időpontban a rendszer mekkora valószínűséggel működik jól.

6. Teljesítmény

Performance

Reméljük szoftverünk a CLP stabilan fogja tudni hozni az alábbi értékeket:

- *Válaszidő (milyen gyorsan reagál egy weblap egy funkció kérésére):* Úgy tervezzük, hogy legkésőbb egy tizedmásodpercig tartson a válaszidő az egyszerűbb feladatok végrehajtásánál, de ez nagyban függ a felhasználói oldalon lévő internet sebességtől, és a hardver architektúrától.
- *Áteresztőképesség (átviteli sebesség):* úgy tervezzük (mivel elég sok adat van) 1000 tranzakció/MP lenne ez az érték.
- *Kapacitás:* Hatalmas kapacításra lenne szükségünk, hogy az összes könyvet, és a rajtuk végzendő funkciókat le tudjuk tárolni.
- *Erőforrás igények:* Mivel a termék webes felületen érhető el egy kiszolgálón keresztül, ezért mindenképp szükség van egy webböngészőre (pl.: Internet Explorer, Mozilla Firefox, stb). Ez akár ingyenesen is beszerezhető, vagy alapértelmezetten telepítve van az operációs rendszerre. Kliens felől semmilyen erőforrást nem vesz majd igénybe, ugyanis nem ő fogja futtatni a szoftvert, hanem a szervergépeken fog. A szerver oldalon sokkal nagyobb erőfeszítésre kell számítani, az egyszerre érkező kérések kezeléséhez.

7. Támogatottság

A későbbi hibakeresés megkönnyítése érdekében, a fejlesztők egységes elnevezési konvenciókat alkalmaznak a kódolás során. Ez nem csak a későbbi hibakeresést, de magát a fejlesztést is könnyebbé, átláthatóbbá teszi.

Továbbá írni fogunk egy olyan, háttérben folyamatosan futó, kis programot, ami folyamatosan figyeli és dokumentálja, hogy mi is történik éppen. Ezt egy körülbelül 10 MB-os szövegfájlba fogja kimenteni, ha kérjük (persze nem a teljes működést, amióta megnyitottuk a weboldalt a publikumnak, hanem csak annyit amennyi belefér a file-ba). Ez azért lesz majd jó, hogy ha esetleg hiba lépne föl (például elfogyott a rendelkezésre álló memória), látni fogjuk majd, hogy hol csúszott meg a programunk működése. A jövőben tervezünk egy ugyanilyen log-ot készíteni a felhasználók számára is, hogy helyileg tudják javítani, ha esetleg valami félrecsúszott (ennek persze csak a mobil alkalmazás esetén lesz értelme).

A fejlesztőknek továbbá segítségére lesz a hamis felhasználók futtatása, amivel át tudnak váltani felhasználói nézetre amikor csak akarnak, és így meg tudják nézni azokat a hibákat, amiket a kliensek jelentettek. Így az ő eszközeiket használva tudják ellenőrizni, hogy tényleges hibáról van-e szó, vagy csak a weboldalhoz nem értésről.

Egy fórum létrehozása is tervbe van véve, ahol kérdések és válaszok alapján fogjuk majd kezelni a bejövő panaszokat. Valamint itt lehet komolyabb hibákról (mint például oldal nem elérhetősége) is jelentést tenni. Itt minden, amit a felhasználók posztolnak az látható lesz társaiknak is, így akár egymást is tudják segíteni, ha valamit nem értenek, ezzel is könnyítve a mi munkánkat. Viszont akinek nem tetszik majd az oldal, azoknak egy e-mail címet is létrehozunk, ahova szintúgy lehet panaszokat küldeni, csak hogy ez nem lesz publikusan látható, mint a weboldal. Végző esetben lesz egy hibajelentő vonal (+06 30 69 69 420) amit fel tudnak hívni a felhasználók, hogy azonnali segítséget kapjanak majd. Az ide beérkezőkre állíthatunk be embereket, de én inkább azt javaslom, hogy egy viszonylag jól működő AI-t alkalmazzunk erre a pozícióra, az emberi erőforrások kímélésére. Ugyanis az AI-t csak betanítani kell, ami igaz lehet egy hosszabb melő, de aztán már nem nagyon kell vele foglalkozni.

8. Tervezési korlátozások

Az alkalmazásunk Java alapon fog működni, amely biztosítja a platformfüggetlenséget. Az alkalmazás bármely operációs rendszer alatt működni fog, csupán egy webböngésző kell hozzá, illetve a mobil alkalmazásunk IOS, Android és Windows Phone alatt is működőképes.

Android fejlesztéshez használt nyelvek:

- Kotlin: Kotlin jelenleg az Android fejlesztés egyik fő nyelve. Kotlin modern, kifejezős, és teljes kompatibilitást biztosít a Java-val.
- Java: Korábban a hivatalos Android fejlesztési nyelv volt, és még mindig sok alkalmazás használja.

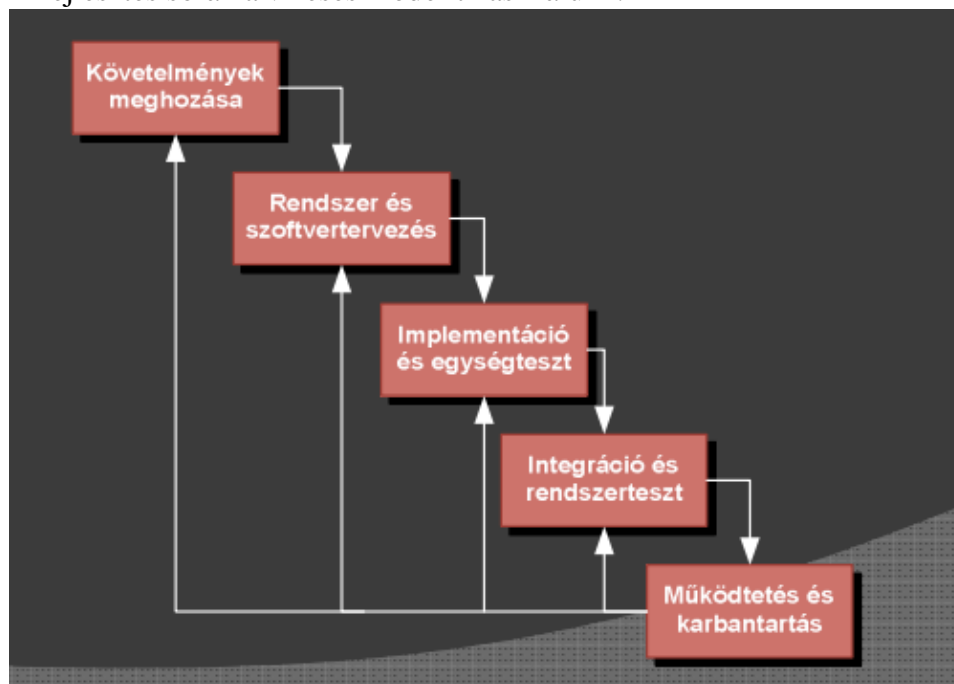
iOS fejlesztés:

- Swift: Swift az Apple által kifejlesztett és támogatott nyelv, amelyet az iOS és macOS alkalmazásokhoz használnak. Gyors és hatékony, és számos modern programozási szempontot figyelembe vesz.

Fejlesztőeszközök:

- Java, Kotlin: IntelliJ
- Swift: Xcode

A fejlesztés során a vízesés modellt használunk.



A vízésés fejlesztési modell egy hagyományos és lineáris projektmenedzsment módszer, amelyet gyakran használnak szoftverfejlesztés és egyéb projekt alapú munkák során. A módszer népszerűségét számos előnye miatt érdemes fontolóra venni:

- **Jól meghatározott szakaszok:** A vízésés modell szigorúan meghatározza a projekt különböző szakaszait, amelyek jól elkülönülnek egymástól. Ez lehetővé teszi, hogy a projekteket könnyen megtervezhető és ellenőrizhető módon kezeljék.
- **Jól definiált követelmények:** A vízésés modell azt követeli, hogy a projekt kezdetén teljesen meghatározzák a követelményeket. Ez segít elkerülni a későbbi módosításokat és problémákat a fejlesztés folyamatában.
- **Kiszámítható ütemezés:** A projekt szakaszait szigorúan definiálják, és minden szakaszt csak akkor kezdnek el, ha a korábbi befejeződött. Ez kiszámítható ütemezést és határidőket eredményez.
- **Jó dokumentáció:** A vízésés modell a projekt minden szakaszában különféle dokumentációkat és tervek igényel, ami segít a projektrészletek és követelmények pontos nyomon követésében.
- **Alapos tesztelés:** A vízésés modell beépíti az alapos tesztelést és minőségellenőrzést a projekt különböző szakaszaiba. Ez segít minimalizálni a hibákat és a problémákat a projekt befejezésekor.
- **Jól megtervezett és strukturált:** A vízésés modell egy jól meghatározott és strukturált megközelítést kínál a projektmenedzsmenthez. Ez segít a projekt szervezésében és irányításában.

Azonban fontos megjegyezni, hogy a vízésés modellnek vannak korlátai és hátrányai is. Például nem rugalmas, és nehezen alkalmazható olyan projektekre, ahol a követelmények gyakran változnak. Emellett a hosszú tervezési és fejlesztési folyamatok miatt nehezebb az ügyfél igényeinek azonnali kielégítése. Ezek a korlátok miatt sok szervezet inkább az agilis fejlesztési módszereket alkalmazza, amelyek jobban alkalmazkodnak a változó követelményekhez és gyorsabban szállítják az eredményeket. A vízésés modell tehát hasznos lehet bizonyos típusú projektek esetében, de nem minden esetben optimális választás.

9. Online dokumentáció és Help rendszer

Online dokumentációra és különböző segítő rendszerekre az esetek többségében szükség van a felhasználók segítése, a könnyebb betanulás, szoftverhasználat, esetleges problémák megoldása, hibák feltárása céljából. Kulcsfontosságú lehet a felhasználók és az adminisztrátorok számára. Ezen rendszerek segítségével könnyebben megérthetik a szoftver működését, beállításait és funkcióit.

A dokumentáció fő tartalmi elemei:

- Telepítési és Frissítési Útmutató: Részletes leírás a szoftver telepítéséről és frissítéséről.
- Alapvető Használati Útmutató
- Specifikációk és Funkciók
- Gyakori Hibák és Megoldások
- Kompatibilitás
- Kapcsolatfelvételi információk

Az online dokumentáció célja, hogy a felhasználó önállóan megértse és használja a szoftvert anélkül, hogy támogatásra lenne szüksége.

Ma, az internet világában szükség van online dokumentációra. Azon belül is kiemelten kell kezelni a program eredményes és könnyű használatát segítő súgó rendszert.

A súgó rendszer segítségét legtöbbször nem a szakemberek, hanem a hétköznapi felhasználók igénylik, így elsősorban az ő igényeiket, és tudásukat szem előtt tartva kell azt elkészíteni. Fontos tehát a világos, lehetőség szerint a szaknyelvtől mentes, mindenki számára érthető fogalmazás, valamint a legkisebb funkcióknak is a lehető legaprólékosabb leírása. Továbbá fontos még a könnyű navigálhatóság a dokumentumon belül. Ezért mind a dokumentum elején található tartalomjegyzékből, mind közvetlenül, az egyes fejezeteken belül található egymásra való hivatkozásokból el lehet érni a kívánt funkció leírását.

10. Felhasznált kész komponensek

A programnak gyakorlatilag a teljes része egyedileg, JAVA nyelven implementált szoftver. Nincs tervbe véve külső komponens használata, csupán a java futtatásához szükséges fordítók.

11. Interfészek

11.1. Felhasználói interfészek

A CPL által, használt szabványok főként a következő területekre vonatkoznak:

1. MARC (Machine-Readable Cataloging) szabvány: A MARC szabvány a könyvtárak által használt katalógusolási formátum, amely lehetővé teszi a könyvek és egyéb könyvtári anyagok gépi olvasását és lekérdezését.
2. Z39.50: A Z39.50 protokoll lehetővé teszi a könyvtári adatbázisok közötti keresést és adatcsere lehetőségét. Ez a szabvány segít az interoperabilitásban és az adatok megosztásában.
3. MARC21: A MARC21 a MARC formátum modernizált változata, amely a MARC adatok elektronikus katalógusolására és megjelenítésére szolgál.
4. Dublin Core Metadata Initiative (DCMI): A DCMI szabvány az egyszerű metaadatok létrehozására és kezelésére szolgál, és hasznos lehet a könyvtári adatbázisokban a tartalomleírás és a keresési metaadatok terén.
5. ISO 2709: Az ISO 2709 szabvány egy formátum a MARC rekordok tárolására és átvitelére. Segíti az adatok importálását és exportálását más könyvtári rendszerekbe.

6. Library of Congress Subject Headings (LCSH): Az LCSH szabvány a könyvtári tárgyszavak és témakategóriák standardizált listáját tartalmazza, amelyek segítik a könyvek és egyéb anyagok címkézését és kategorizálását.
7. OpenURL: Az OpenURL egy protokoll, amely lehetővé teszi a könyvtárak számára, hogy hivatkozásokat hozzanak létre és kezeljenek az elektronikus forrásokhoz, például online cikkekhez és e-könyvekhez.

Ezen szabványok közül néhány a könyvtári adatbázisok katalógizálásához és metaadatkezeléséhez kapcsolódik, míg mások az adatcserére és az interoperabilitásra vonatkoznak. A konkrét szabványok kiválasztása és alkalmazása attól függ, hogy milyen típusú könyvtár adatbázis szoftvert használ, és milyen igények vannak az adatkezelésre és az adatcserére a könyvtári környezetben. Az intézményeknek és szoftvertervezőknek érdemes alaposan megfontolniuk ezeket a szabványokat a megfelelő működés és adatkezelés biztosítása érdekében.

A weboldal megnyitása után egy könnyen kezelhető, letisztult felület jelenik meg a felhasználók számára. A felhasználók regisztrálhatnak új fiókot vagy bejelentkezhetnek meglévő fiókjaikba. Az főoldalon látványt tekintve egységes, de az elérhető funkciók a különböző szerepköröknek megfelelően változnak. A különböző menüpontokon keresztül érhetik el a felhasználók a különböző funkciókat.

11.2. Hardware interfészek

Az alkalmazás nem igényel külön hardvert. Bármely eszközről futtatható, amely rendelkezik, internet eléréssel és webböngészővel.

11.3. Software interfészek

A CPL nem igényel kommunikációt külső alkalmazásokkal, szerverekkel. A webes felület és a könyvtár raktárkészlet kommunikáció szükséges egyedül a zökkenőmentes használathoz.

11.4. Kommunikációs interfészek

Az alkalmazás működéséhez nem szükséges speciális hardver. Bármely személyi számítógépen futtatható, amely kompatibilis az alkalmazás által elvárt operációs rendszerekkel, tehát nem igényel speciális vagy erőforrásigényes számítógépet. Ezenkívül mindössze egy webböngésző telepítése és internetkapcsolat szükséges a használatához, így könnyedén elérhető és futtatható bárhol, ahol van internetelérés. Ezáltal nagyfokú rugalmasságot nyújt a felhasználóknak, és lehetővé teszi az alkalmazás kényelmes és széleskörű használatát.

12. Alkalmazott szabványok

12.1. Kötelezően alkalmazandó szabványok

A törvény szerint meghatározott egy évig tartó jótállást vállalunk a szoftverre. Ebben az időszakban garantáljuk, a program hibamentes futását. A gyártó nem vállal felelősséget olyan károkért, amelyeket nem a szoftver rendeltetésszerű használatának eltérő módja vagy vírusátadás okoz.

12.2. Választás alapján alkalmazott szabványok

A fejlesztők vagy a felhasználók által valamilyen okból önként vállalt előírások.

Adatbiztonság: Biztosítjuk a megrendelőinknek és a felhasználóinknak az adataik védelmét, titkosítását. Garantáljuk az adatok fizikai és hálózati biztonságát az adatbiztonsági kockázatok elkerülése érdekében.

Testreszabhatóság: A fejlesztői csapatunk a felhasználói igényeknek és a megrendelőknek egyaránt testre szabja a funkciókat és a kezelőfelületet.

13. Mellékletek

Appendix

Főleg hosszabb dokumentumok esetén ide kerülhet minden olyan információ, ami a dokumentum áttekinthetőségét segíti (pl. tárgymutató, irodalomjegyzék, egyéb források megjelölése), illetve a fejlesztés egyéb, kapcsolódó dokumentumaira való hivatkozás.